Scientific Item Recommendation Using a Citation Network

Xu Wang[1,2(✉)], Frank van Harmelen[1,2], Michael Cochez[1,2], and Zhisheng Huang[1]

[1] Vrije Universiteit Amsterdam, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands
{xu.wang,frank.van.harmelen,m.cochez,z.huang}@vu.nl
[2] Discovery Lab, Elsevier, Amsterdam, The Netherlands

Abstract. Scientific items (such as papers or datasets) discovery and reuse is crucial to support and improve scientific research. However, the process is often tortuous, and researchers end up using less than ideal datasets. Search engines tailored to this task are useful, but current systems only support keyword searches. This hinders the process, because the user needs to imagine what kind of keywords would give useful datasets. In this paper, we investigate a new technique to recommend scientific items (paper or datasets). This technique uses a graph consisting of scientific papers, the corresponding citation network, and datasets used in these works as background information for its recommendation. Specifically, a link-predictor is trained which is then used to infer useful datasets for the paper the researcher is working on. As an input, it uses the co-author information, citation information, and the already used datasets. To compare different scientific items recommendation approaches fairly and to prove their efficiency, we created a new benchmark. This benchmark includes more than three million scientific items to evaluate the performance of recommendation approaches. We experiment with a variety of methods and find that an ensemble technique which uses link prediction on the citation network yields a precision of nearly 70%.

Keywords: Data discovery · Data reuse · Scientific items recommendation · Recommendation benchmark · Link prediction

1 Introduction

Data discovery and reuse play an essential role in helping scientific research by supporting to find data [4,19]. Researchers typically reuse datasets from colleagues or collaborators, and the credibility of such datasets is critical to the scientific process [11,25]. Datasets sourced from a network of personal relationships (colleagues or collaborators) can carry limitations as they tend only to recommend datasets that they themselves find helpful [2]. However, due to the research variability, one person's noisy data may be another person's valuable data. Also, datasets retrieved from relational networks can be limited to certain research areas.

G. Memmi et al. (Eds.): KSEM 2022, LNAI 13369, pp. 469–484, 2022.
https://doi.org/10.1007/978-3-031-10986-7_38

As an emerging dataset discovery tool, a dataset search engine can help researchers to find datasets of interest from open data repositories. Moreover, due to the increasing number of open data repositories, many dataset search engines, such as Google Dataset Search [3] and Mendeley Data[1], cover more than ten million datasets. While dataset search engines bring convenience to researchers, they also have certain limitations. Similar to general search engines, such dataset search engines require the researcher to provide keywords to drive the search; filtering, ranking, and returning all datasets based on the given keywords. In order to use a dataset search engine, researchers need to summarize the datasets they are looking for into these keywords, with the risk that they do not cover all the desired properties, and that unexpected but relevant datasets will be missed. Thus, the standard pathway "scientific items → keywords → scientific items sets"[2] used by existing dataset search engines has inherent limitations.

This paper proposes a recommendation method based on entity vectors trained on citation networks. This approach is a solution for data discovery following the more direct "scientific items → scientific items" pathway. Because our approach does not require converting scientific items (papers and datasets) into keywords, we can avoid the earlier drawbacks. Furthermore, we combine this new recommendation method with existing recommendation methods into an integrated ensemble recommendation method. This paper also provides a benchmark corpus for scientific item recommendation and a benchmark evaluation test. By performing benchmark tests on randomly selected scientific items from this benchmark corpus, we conclude that our integrated recommendation method using citation network entity embedding can obtain a precision rate of about 70%.

Specifically, in this paper, we study three research questions:

- Will a citation network help in scientific item discovery?
- Can we do dataset discovery purely by link prediction on a citation network?
- Will the addition of citation-network-link-prediction help for scientific item discovery?

The main contributions of this paper are: 1) we propose a method for recommending scientific items based on entity embedding in an academic citation graph, 2) we propose a benchmark corpus and evaluation test for scientific items recommendation methods, 3) we identify an ensemble method that has high precision for scientific items recommendation, and 4) we provide the pretrained entity embeddings for our large-scale academic citation network as an open resource for re-use by others.

2 Related Work

Data reuse aims to facilitate replication of scientific research, make scientific assets available to the public, leverage research investment, and advance research

[1] https://data.mendeley.com/.

[2] We use the term "scientific items" to refer to both papers and datasets.

and innovation [19]. Many current works focus on supporting and bringing convenience to data reuse. Wilkinson et. al. provided FAIR guiding principles to support scientific data reuse [28]. Pierce et. al. provided data reuse metrics for scientific data so that researchers can track how the scientific data is used or reused [22]. Duke and Porter provided a framework for developing ethical principles for data reuse [10]. Faniel et. al. provided a model to examine the relationship between data quality and user satisfaction [12].

Dataset recommendation is also a popular research trend in recent years. Farber and Leisinger recommended suitable dataset for given research problem description [14]. Patra et. al. provided an Information retrieval (IR) paradigm for scientific dataset recommendation [20]. Altaf et. al. recommended scientific dataset based on user's research interests [1]. Chen et. al. proposed a three-layered network (composed of authors, papers and datasets) for scientific dataset recommendation [5].

3 Link Prediction with Graph Embedding on a Citation Network

The link prediction training method we use is KGlove [7]. KGlove finds statistics of co-occurrences of nodes in random walks, using personalized page rank. Then Glove [21] is used to generate entity embeddings from the co-occurrence matrix. In this paper, we apply KGlove on 638,360,451 triples of the Microsoft Academic Knowledge Graph (MAKG) [13] citation network (containing 481,674,701 nodes) to generate a co-occurrence matrix of the scientific items. Then we use the Glove method on this co-occurence matrix to obtain the scientific entity (item) embeddings. The trained embeddings are made available for future work[3]. After training the entity embedding based on the MAKG citation network, we perform link predictions between scientific items (papers and/or datasets) by a similarity metric in the embedding space. We use cosine similarity, which is the most commonly used similarity for such embeddings.

Definition 1 (Link Prediction for scientific items with Entity Embedding). *Let* $E = \{e_1, e_2, ...\}$ *be a set of scientific entities (also known as scientific items). Let* emb *be an embedding function for entities such that* emb(e) *is the embedding of entity* $e \in E$*, and* emb(e) *is a one-dimensional vector of a given length.*
Let $\cos : (a, b) \rightarrow [0, 1]$ *be a function such that* $\cos(a, b) = \frac{\text{emb}(a) \cdot \text{emb}(b)}{||\,\text{emb}(a)|| \cdot ||\,\text{emb}(b)||}$ *where* $a, b \in E$.
Given a threshold t*, we define* Link prediction with Entity Embedding *in* E *as a function* $LP_E : E \rightarrow 2^E$ *where* $LP_E(e_s) = \{r_1, r_2, ..., r_n | \forall i = 1 \ldots n, \cos(e_s, r_i) < t\}$.

[3] https://zenodo.org/record/6324341.

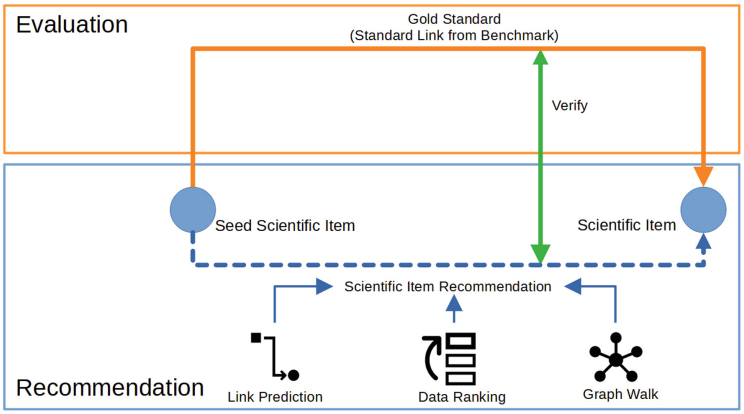

Fig. 1. Pipeline of Scientificset Data Recommendation and Evaluation Benchmark.

4 Dataset Recommendation Methods

In this section we use the previous definition of link prediction to introduce two new dataset recommendation methods, as well as three methods from our previous work. We also propose an open-access scientific items recommendation evaluation benchmark, including corpus and evaluation pipeline (Fig. 1).

4.1 Dataset Recommendation Methods

The dataset recommendation methods in this section use a combination of link-prediction and ranking approaches to recommend a recommended scientific item based on given scientific items.

Data Recommendation with Link Prediction Using a Citation Network. This scientific entity (item) recommendation method is based on Definition 1, where a set of entities is returned such that the cosine distance between these entities and the given entity is smaller than a threshold t. Based on the list of scientific items returned by the link prediction algorithm, the recommendation method considers only the TOP-n results of that list, with the value of n to be chosen as a parameter of the method. Formally, this is defined as follows:

Definition 2 (Top-n scientific entity (items) Recommendation with Link Prediction). *Let $E = \{e_1, e_2, ...\}$ be a set of scientific entities (also known as scientific items). Let LP_E be a link prediction function using embeddings in E (see Definition 1). Top-n Scientific entity recommendation with link prediction using embedding is a function $DRLP_E^n$, which maps an entity e_s to $(r_1, \ldots r_m)$ which is the longest ordered list of $m <= n$ pairwise distinct elements of $LP_E(e_s)$ where $\forall i = 1 \ldots m-1, \cos(e_s, r_i) <= \cos(e_s, r_{i+1})$.*

In words, this function maps an entity (scientific item) to a list of at most n other entities (scientific items) which are closest to it in the embedded space, ordered by the distance.

We can now combine this general definition with a specific embedding function emb to create a specific link-prediction-based recommendation method. In particular, we use KGloVe embeddings from the MAKG citation network to create a recommendation method based on link prediction from a citation network.

Scientific Items Recommendation with BERT-based Link Prediction. The method from the previous subsection used the embeddings computed on the citation graph to determine similarity between data items. This is a plausible choice, since we can expect the MAKG citation graph to give us a reasonable signal for similarity in the scientific domain: it captures the scientific relationships between items in the science domain. In contrast to this, we also experimented with using other models to compute the similarity between items. In particular, we used the pretrained BERT model [9,23] as an example of a cross-domain model to see if such a generic pretrained model would also suffice to compute the similarity metric that is the basis for our link-prediction-based recommendation algorithm. The pretrained BERT model used in this paper is the *all-mpnet-base-v2* model from the SentenceTransformers Python library[4]. Such BERT-based link prediction for scientific items is obtained by applying the pretrained BERT model to the descriptive metadata of the scientific items to obtain the BERT embedding of scientific items. Such metadata consists of the title of the dataset and a short text that accompanies the dataset. Then, we apply the BERT embedding of the scientific items to Definition 2 to do scientific items recommendations.

Scientific Items Recommendation with BM25-based Data Ranking. BM25-based Data Ranking is the recommendation approach provided in our previous paper [27]. Given a seed scientific item, we rank the list of candidate recommended scientific items using the popular BM25 method from information retrieval according to the descriptive metadata of the scientific items (consisting of title and textual description), where a higher ranking position means a better recommendation [24].

Scientific Items Recommendation with Graph Walk. The co-author network-based graph walk method is also a scientific items recommendation method that we have previously proposed in [26]. Such a graph walk on a co-author network performs the recommendation task according to the "scientific items → author → co-author network → author → scientific items" pathway. In order to reduce the number of candidate recommendations we only consider items connected to authors within an n-hop distance to the author of the seed data item in the co-author network.

[4] https://www.sbert.net/docs/pretrained_models.html.

Table 1. Statistics of benchmark corpus

	Number of items	Number of links
All	3,227,206	15,979,748
Paper-Paper	2,909,755	14,391,413
Dataset-Dataset	1,544	2,335

Dataset Recommendation with Pre-trained Author Embedding. Similar to the method based on citation-based embeddings, we have proposed in earlier work [26] a recommendation method for scientific items based on pre-trained co-authorship embeddings. This approach is similar to our proposed method using embeddings from the MAKG citation network (Definition 1), but uses embeddings computed from the MAKG co-author network instead.

5 Scientific Items Recommendation Benchmark

To evaluate the performance scientific items recommendation methods, we propose here an open-source generalized benchmark corpus and process for scientific items recommendation. scientific items in general can be publications, datasets, graphs, tables, geographic data, etc.

5.1 Benchmark Corpus

The benchmark corpus is an HDT/RDF graph [15,18] stored as triples of the form "[scientific item] [link] [scientific item]." The scientific items are the intersection of scientific items in ScholeXplorer[5] and MAKG (Microsoft Academic Knowledge Graph). This intersection is computed by matching the DOI of scientific items (datasets and/or papers) between ScholeXplorer and MAKG. We have chosen to represent all the scientific items by the identificatier used in the Microsoft Academic Graph (MAG). With help of these MAG identifiers, the information (such as title, providers, publishers, or creators) of scientific items is easily accessible in MAKG. The bi-directional links between these items are from ScholeXplorer and all the links are provided by data sources managed by publishers, data centers, or other organizations.

In Table 1, we show the statistics of our benchmark corpus. There are more than 3 million items and more than 15 million bi-directional links between them. We provide the data subset with only bi-directional links between scientific *papers*, consisting of 2.9 million scientific papers and 14.3 million links between them. We also provide the data subset of only the bi-directional links between scientific *datasets*, with 1,544 scientific items and 2,335 million links between them. We have made this corpus available at https://zenodo.org/record/6386897.

[5] https://scholexplorer.openaire.eu/.

Table 2. Statistics of experiments

	Number of seeds	Number of candidates
Exp1	12	115
Exp2	109	27,242
Exp3	181	28,960

5.2 Benchmark Evaluation

The goal of our benchmark is to evaluate the performance of scientific item recommendation methods on all datasets in the benchmark corpus, with the option to only use a randomly selected subset. We use the F1-measure method [6] to evaluate the performance of recommendation methods on reconstruction of bi-directional links between scientific items. The F1-measure method consists of three evaluation metrics: recall, precision and F1-score. Recall is the percentage of recommendations (i.e. links as given in the dataset that start from the seed data) that the recommendation method can recommend. Precision is the percentage of scientific items recommended by the recommendation method that is correct (i.e., present in the standard). Finally, the F1-score is the harmonic mean of recall and precision.

6 Experiments and Results

This section will present the setup and results of our experiments on the proposed recommendation methods from Sect. 4 using the evaluation benchmark from Sect. 5. The implementation of recommendation methods and the code of all experiments could be found at https://github.com/XuWangVU/datarecommend.

6.1 Experimental Setup

We set up three evaluation experiments using three sets of data randomly selected from the benchmark corpus. The statistics of the selected data are shown in Table 2. For each seed scientific item, we look for recommendations among all the candidate scientific items and return a sorted subset of these candidates.

The recommendation methods evaluated in the experiments comprise the five methods described in Sect. 4. Beyond these single methods, we also tested ensemble methods by combining multiple methods to make recommendations. All methods (including the ensemble methods) fall into two types of pathway-based categories: pathways with author and pathway without authors. All the methods (including the ensemble methods) used in our experiments can be found in Table 3.

We use thresholds for two methods: a distance threshold for graph walks and a threshold for similarity between author embeddings. The distance threshold

Table 3. Scientific items recommendation methods used for experiments.

Approach name	Link prediction (Citation)	Data ranking (BM25)	Link prediction (BERT)	Graph walk	Author embedding
Citation	X				
BERT			X		
BM25		X			
Citation + BM25	X	X			
Citation + BERT	X		X		
BERT + BM25		X	X		
Citation + BM25 + BERT	X	X	X		
Pure walk				X	
Hop(n) + Embed(T)				X	X
Hop(n) + Embed(T) + Citation	X			X	X
Hop(n) + Embed(T) + BM25		X		X	X
Hop(n) + Embed(T) + BERT			X	X	X
Hop(n) + Embed(T) + BERT + BM25		X	X	X	X
Hop(n) + Embed(T) + BM25 + Citation	X	X		X	X
Hop(n) + Embed(T) + BERT + Citation	X		X	X	X
Hop(n) + Embed(T) + All	X	X	X	X	X

for graph walks is the maximum number of hops that make up a graph walk. For example, hop1 means that only authors with a distance of 1 from the given author are considered. The author embedding similarity threshold means that only authors with an embedding similarity greater than or equal to the threshold with the given author are considered.

Each recommendation method is assigned a parameter. For the graph walk method, we use the parameter of hop1, hop2, or hop3, to represent the distance threshold used for graph walk. For the similarity method between pretrained MAKG author embeddings, we use similarity threshold parameters ranging from 0.3 to 0.7, increasing in steps of 0.1. For the BM25-based ranking method, we use the parameter $p_{bm25} = 2 * outdegree(seed)$, where $outdegree(seed)$ is the number of scientific items linked from the seed in the benchmark corpus. In other words, we will only consider the top p_{bm25} results in the list returned by the ranking method. For both link prediction methods using citation network embeddings and BERT-based link prediction methods, we use a parameter of 0.8, which means we only consider the top 80% of the sorted lists returned by both methods.

6.2 Experimental Results

Table 4 show the results of the scientific items recommendation methods which do not consider authors in the pathway, while Tables 5, 6 and 7 show the results of methods considering authors. We use color-coding of the cells to indicate different ranges of values: Red means relative poor performance in comparison with related settings; green code means outstanding performance in comparison; and yellow means average performance.

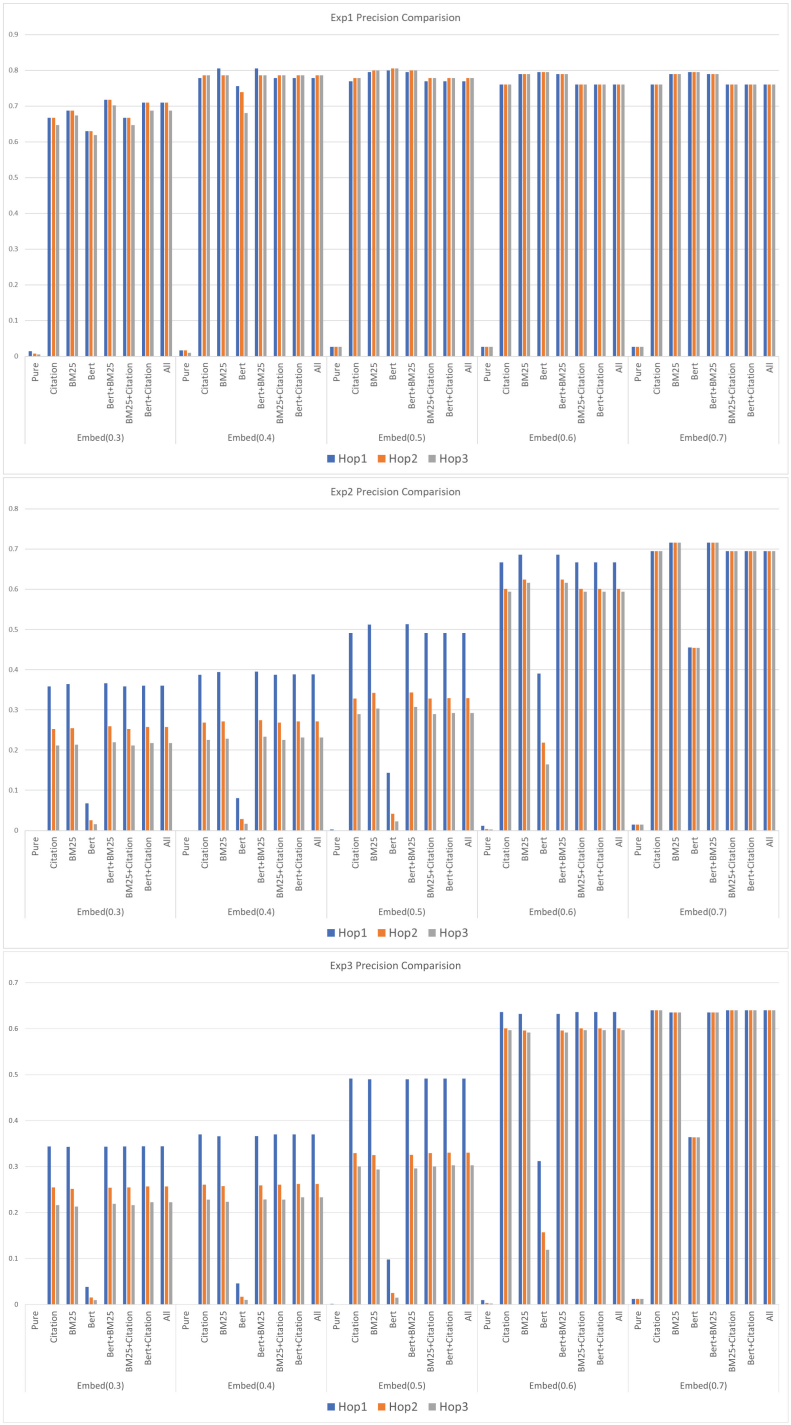

Fig. 2. Precision comparison in Experiment 1, Experiment 2, and Experiment 3.

Table 4. Results of Experiment(EXP) 1, 2 & 3 without graph walk and author embedding.

	EXP1			EXP2			EXP3		
	R	P	F1	R	P	F1	R	P	F1
Citation	0,12458	0,22561	0,16052	0,28681	0,17941	0,22074	0,2986	0,1869	0,2299
Bert	0,15825	0,19583	0,17505	0,96881	0,0127	0,02507	0,9700	0,0076	0,0151
BM25	0,15488	0,2201	0,18182	0,35851	0,17925	0,239	0,3720	0,1860	0,2480
Citation+BM25	0,12458	0,22561	0,16052	0,28681	0,17941	0,22074	0,2986	0,1869	0,2299
Citation+Bert	0,12458	0,27007	0,17051	0,28681	0,18736	0,22665	0,2986	0,1937	0,2350
Bert+BM25	0,15488	0,27381	0,19785	0,35851	0,18738	0,24612	0,3720	0,1930	0,2542
Citation+BM25+Bert	0,12458	0,27007	0,17051	0,28681	0,18736	0,22665	0,2986	0,1937	0,2350

In the experiments which do not consider authors, we found that recall, precision, and F1-score were usually not high, except for the method which only uses BERT, where we could obtain a recall of over 0.95. However, this situation does not achieve sufficiently high precision rates.

When the author network is taken into consideration, the precision rate improves considerably, and in some integrated methods, we achieve precision results of 0.7 or even 0.8. Unfortunately, these high precision rates come with a decreased recall rate, which means that the methods return few, but often correct recommendations.

This behavior, i.e., high precision rates at relative low recall, is typical and sufficient for recommendation engines. Hence, we explore these results in more detail. A comparison of the precision rates of the different methods can be found in Fig. 2. For experiment 1, we observe little variability, likely due to the small data size. For experiments 2 and 3, however, the precision rate increases with a higher distance threshold for the graph walk or with a higher threshold for the author embedding similarity.

Based on the comparison of the results of the different methods in Tables 5, 6 and 7 and Fig. 2, we can conclude that all recommendation methods that use data ranking (BM25) or link prediction (Citation Embedding) have a high precision on our scientific items recommendation benchmark experiments when using graph walking and author embedding similarity methods in an ensemble of methods.

7 Conclusion and Discussion

In this paper, we have investigated the use of a large scale citation network for the purposes of recommending scientific items, on the basis of a given scientific item by the user, according to the well-known paradigm “if you like this dataset, you might also like these other datasets”. The method uses low-dimensional vector space embeddings computed from the citation graph in order to compute the cosine similarity between datasets as the basis for its recommendations. By itself, this method performed unsatisfactorily on our benchmark under a variety of experimental settings.

We therefore also studied the behaviour of this method in an ensemble with a number of other methods: recommendations based on n-hops walks in a co-author graph ($n = 1, 2, 3$), recommendations based on embeddings computed over this co-author graph, recommendations based on the BERT large language model, and the BM25 method from information retrieval. We studied a large variety of the most promising combinations of methods under different experimental settings. In our largest experimental setting, the ensemble methods that used the embeddings from the citation network outperformed those that didn't, with a precision of 0.64 under a variety of settings. This acceptable precision in a recommendation setting comes at the price of a low recall, a behaviour that is typical in recommendation engines.

This allows us to succinctly answer the research questions we formulated in the introduction of this paper:

- Will a citation network help in dataset discovery? Answer: yes
- Can we do dataset discovery purely by link prediction on a citation network? Answer: no
- Will the addition of citation-network-link-prediction help for dataset discovery? Answer: yes

We performed our experiments on a newly constructed benchmark set, using the KGlove method for training scientific entity (item) embeddings from the Microsoft Academic Knowledge Graph, containing a citation network of 100 million edges. We have made this benchmark corpus available online.

The methods that we designed and evaluated in this paper are clearly not the final word on how to recommend scientific items. Likely, the results can be improved not only by using tuning parameters to specific datasets, but also by adding other existing applicable methods. Also, the dataset could be expended. We have used both citation and co-author networks as signals for academic similarity, but also other academic networks exist. Including those is subject of future work.

The link prediction mentioned in this paper uses pre-trained embedding models. One drawback of this type of models is that this requires an embedding for each entity in the graph, and hence many existing models do not scale well enough. In the future several approaches could be investigated to overcome, one option is to use a model which can work in an inductive setting, based on the description, or even the content of the datasets. An example of such a method is BLP [8]. To reduce the number of embeddings, we could also use a model which only keeps embeddings for some entities in the graph, like NodePiece [16]. Another direction could be to attempt scaling models using summarization, as was done in [17].

Acknowledgments. This work was funded in part by Elsevier's Discovery Lab (https://discoverylab.ai/). This work was also funded by the Netherlands Science Foundation NWO grant nr. 652.001.002 which is also partially funded by Elsevier. The first author is funded by the China Scholarship Council (CSC) under grant nr. 201807730060. Part of this work was inspired by discussions with other members of the discovery lab, like Daniel Daza and Dimitrios Alivanistos.

A Detailed Results of the Different Experiments

Table 5. Results of Experiment 1 with graph walk and author embedding.

Exp1	Hop1			Hop2			Hop3		
	R	P	F1	R	P	F1	R	P	F1
Pure walk	0.2189	0.0139	0.0261	0.2189	0.0048	0.0095	0.2189	0.0047	0.0092
Embed(0.3)	0.2189	0.0139	0.0262	0.2189	0.0071	0.0138	0.2189	0.0052	0.0101
Embed(0.3)+Citation	0.0741	0.6667	0.1333	0.0741	0.6667	0.1333	0.0741	0.6471	0.1329
Embed(0.3)+BM25	0.1111	0.6875	0.1913	0.1111	0.6875	0.1913	0.1111	0.6735	0.1908
Embed(0.3)+BERT	0.1145	0.6296	0.1937	0.1145	0.6296	0.1937	0.1145	0.6182	0.1932
Embed(0.3)+BERT+BM25	0.1111	0.7174	0.1924	0.1111	0.7174	0.1924	0.1111	0.7021	0.1919
Embed(0.3)+BM25+Citation	0.0741	0.6667	0.1333	0.0741	0.6667	0.1333	0.0741	0.6471	0.1329
Embed(0.3)+BERT+Citation	0.0741	0.7097	0.1342	0.0741	0.7097	0.1342	0.0741	0.6875	0.1337
Embed(0.3)+All	0.0741	0.7097	0.1342	0.0741	0.7097	0.1342	0.0741	0.6875	0.1337
Embed(0.4)	0.2189	0.0166	0.0308	0.2189	0.0162	0.0301	0.2189	0.0096	0.0184
Embed(0.4)+Citation	0.0707	0.7778	0.1296	0.0741	0.7857	0.1354	0.0741	0.7857	0.1354
Embed(0.4)+BM25	0.1111	0.8049	0.1953	0.1111	0.7857	0.1947	0.1111	0.7857	0.1947
Embed(0.4)+BERT	0.1145	0.7556	0.1988	0.1145	0.7391	0.1983	0.1145	0.6800	0.1960
Embed(0.4)+BERT+BM25	0.1111	0.8049	0.1953	0.1111	0.7857	0.1947	0.1111	0.7857	0.1947
Embed(0.4)+BM25+Citation	0.0707	0.7778	0.1296	0.0741	0.7857	0.1354	0.0741	0.7857	0.1354
Embed(0.4)+BERT+Citation	0.0707	0.7778	0.1296	0.0741	0.7857	0.1354	0.0741	0.7857	0.1354
Embed(0.4)+All	0.0707	0.7778	0.1296	0.0741	0.7857	0.1354	0.0741	0.7857	0.1354
Embed(0.5)	0.2121	0.0260	0.0463	0.2155	0.0261	0.0465	0.2155	0.0261	0.0465
Embed(0.5)+Citation	0.0673	0.7692	0.1238	0.0707	0.7778	0.1296	0.0707	0.7778	0.1296
Embed(0.5)+BM25	0.1044	0.7949	0.1845	0.1077	0.8000	0.1899	0.1077	0.8000	0.1899
Embed(0.5)+BERT	0.1077	0.8000	0.1899	0.1111	0.8049	0.1953	0.1111	0.8049	0.1953
Embed(0.5)+BERT+BM25	0.1044	0.7949	0.1845	0.1077	0.8000	0.1899	0.1077	0.8000	0.1899
Embed(0.5)+BM25+Citation	0.0673	0.7692	0.1238	0.0707	0.7778	0.1296	0.0707	0.7778	0.1296
Embed(0.5)+BERT+Citation	0.0673	0.7692	0.1238	0.0707	0.7778	0.1296	0.0707	0.7778	0.1296
Embed(0.5)+All	0.0673	0.7692	0.1238	0.0707	0.7778	0.1296	0.0707	0.7778	0.1296
Embed(0.6)	0.2088	0.0270	0.0478	0.2088	0.0270	0.0478	0.2088	0.0270	0.0478
Embed(0.6)+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.6)+BM25	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791
Embed(0.6)+BERT	0.1044	0.7949	0.1845	0.1044	0.7949	0.1845	0.1044	0.7949	0.1845
Embed(0.6)+BERT+BM25	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791
Embed(0.6)+BM25+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.6)+BERT+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.6)+All	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.7)	0.2088	0.0270	0.0478	0.2088	0.0270	0.0478	0.2088	0.0270	0.0478
Embed(0.7)+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.7)+BM25	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791
Embed(0.7)+BERT	0.1044	0.7949	0.1845	0.1044	0.7949	0.1845	0.1044	0.7949	0.1845
Embed(0.7)+BERT+BM25	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791	0.1010	0.7895	0.1791
Embed(0.7)+BM25+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.7)+BERT+Citation	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180
Embed(0.7)+All	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180	0.0640	0.7600	0.1180

Table 6. Results of Experiment 2 with graph walk and author embedding.

Exp2	Hop1			Hop2			Hop3		
	R	P	F1	R	P	F1	R	P	F1
Pure walk	0.2230	0.0005	0.0009	0.4425	0.0001	0.0003	0.6225	0.0001	0.0002
Embed(0.3)	0.1903	0.0006	0.0011	0.3838	0.0002	0.0003	0.5330	0.0001	0.0002
Embed(0.3)+Citation	0.0635	0.3581	0.1079	0.1230	0.2521	0.1654	0.1665	0.2115	0.1863
Embed(0.3)+BM25	0.0813	0.3639	0.1329	0.1554	0.2540	0.1928	0.2099	0.2129	0.2114
Embed(0.3)+BERT	0.1886	0.0680	0.0999	0.3804	0.0258	0.0483	0.5283	0.0158	0.0307
Embed(0.3)+BERT+BM25	0.0813	0.3658	0.1330	0.1554	0.2588	0.1942	0.2099	0.2193	0.2145
Embed(0.3)+BM25+Citation	0.0635	0.3581	0.1079	0.1230	0.2521	0.1654	0.1665	0.2115	0.1863
Embed(0.3)+BERT+Citation	0.0635	0.3602	0.1080	0.1230	0.2569	0.1664	0.1665	0.2178	0.1887
Embed(0.3)+All	0.0635	0.3602	0.1080	0.1230	0.2569	0.1664	0.1665	0.2178	0.1887
Embed(0.4)	0.1258	0.0008	0.0016	0.2525	0.0002	0.0004	0.3279	0.0001	0.0002
Embed(0.4)+Citation	0.0424	0.3868	0.0765	0.0821	0.2680	0.1257	0.1056	0.2258	0.1439
Embed(0.4)+BM25	0.0547	0.3942	0.0961	0.1042	0.2710	0.1505	0.1337	0.2279	0.1685
Embed(0.4)+BERT	0.1246	0.0804	0.0977	0.2506	0.0288	0.0517	0.3255	0.0163	0.0310
Embed(0.4)+BERT+BM25	0.0547	0.3952	0.0962	0.1042	0.2744	0.1511	0.1337	0.2335	0.1700
Embed(0.4)+BM25+Citation	0.0424	0.3868	0.0765	0.0821	0.2680	0.1257	0.1056	0.2258	0.1439
Embed(0.4)+BERT+Citation	0.0424	0.3878	0.0765	0.0821	0.2715	0.1261	0.1056	0.2313	0.1450
Embed(0.4)+All	0.0424	0.3878	0.0765	0.0821	0.2715	0.1261	0.1056	0.2313	0.1450
Embed(0.5)	0.0645	0.0021	0.0040	0.0996	0.0004	0.0008	0.1092	0.0002	0.0004
Embed(0.5)+Citation	0.0225	0.4909	0.0430	0.0331	0.3279	0.0602	0.0358	0.2892	0.0637
Embed(0.5)+BM25	0.0302	0.5123	0.0571	0.0438	0.3422	0.0777	0.0475	0.3034	0.0822
Embed(0.5)+BERT	0.0639	0.1437	0.0884	0.0988	0.0420	0.0589	0.1083	0.0227	0.0375
Embed(0.5)+BERT+BM25	0.0302	0.5125	0.0571	0.0438	0.3435	0.0777	0.0475	0.3070	0.0823
Embed(0.5)+BM25+Citation	0.0225	0.4909	0.0430	0.0331	0.3279	0.0602	0.0358	0.2892	0.0637
Embed(0.5)+BERT+Citation	0.0225	0.4912	0.0430	0.0331	0.3292	0.0602	0.0358	0.2927	0.0638
Embed(0.5)+All	0.0225	0.4912	0.0430	0.0331	0.3292	0.0602	0.0358	0.2927	0.0638
Embed(0.6)	0.0462	0.0113	0.0182	0.0478	0.0039	0.0072	0.0480	0.0022	0.0043
Embed(0.6)+Citation	0.0178	0.6667	0.0347	0.0179	0.6007	0.0348	0.0180	0.5941	0.0349
Embed(0.6)+BM25	0.0242	0.6861	0.0467	0.0245	0.6241	0.0471	0.0245	0.6155	0.0471
Embed(0.6)+BERT	0.0457	0.3905	0.0818	0.0473	0.2188	0.0778	0.0475	0.1642	0.0737
Embed(0.6)+BERT+BM25	0.0242	0.6861	0.0467	0.0245	0.6241	0.0471	0.0245	0.6155	0.0471
Embed(0.6)+BM25+Citation	0.0178	0.6667	0.0347	0.0179	0.6007	0.0348	0.0180	0.5941	0.0349
Embed(0.6)+BERT+Citation	0.0178	0.6667	0.0347	0.0179	0.6007	0.0348	0.0180	0.5941	0.0349
Embed(0.6)+All	0.0178	0.6667	0.0347	0.0179	0.6007	0.0348	0.0180	0.5941	0.0349
Embed(0.7)	0.0452	0.0145	0.0219	0.0452	0.0144	0.0219	0.0452	0.0144	0.0219
Embed(0.7)+Citation	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343
Embed(0.7)+BM25	0.0239	0.7157	0.0463	0.0239	0.7157	0.0463	0.0239	0.7157	0.0463
Embed(0.7)+BERT	0.0447	0.4549	0.0814	0.0447	0.4543	0.0814	0.0447	0.4543	0.0814
Embed(0.7)+BERT+BM25	0.0239	0.7157	0.0463	0.0239	0.7157	0.0463	0.0239	0.7157	0.0463
Embed(0.7)+BM25+Citation	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343
Embed(0.7)+BERT+Citation	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343
Embed(0.7)+All	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343	0.0176	0.6950	0.0343

Table 7. Results of Experiment 3 with graph walk and author embedding.

Exp3	Hop1			Hop2			Hop3		
	R	P	F1	R	P	F1	R	P	F1
Pure walk	0.2582	0.0003	0.0007	0.4450	0.0001	0.0002	0.5955	0.0001	0.0001
Embed(0.3)	0.2235	0.0004	0.0007	0.3973	0.0001	0.0002	0.5276	0.0001	0.0001
Embed(0.3)+Citation	0.0827	0.3440	0.1334	0.1331	0.2548	0.1748	0.1706	0.2165	0.1908
Embed(0.3)+BM25	0.1041	0.3431	0.1597	0.1650	0.2518	0.1994	0.2104	0.2130	0.2117
Embed(0.3)+BERT	0.2215	0.0385	0.0657	0.3941	0.0155	0.0298	0.5234	0.0097	0.0191
Embed(0.3)+BERT+BM25	0.1041	0.3434	0.1597	0.1650	0.2541	0.2001	0.2104	0.2193	0.2147
Embed(0.3)+BM25+Citation	0.0827	0.3440	0.1334	0.1331	0.2548	0.1748	0.1706	0.2165	0.1908
Embed(0.3)+BERT+Citation	0.0827	0.3442	0.1334	0.1331	0.2569	0.1753	0.1706	0.2227	0.1932
Embed(0.3)+All	0.0827	0.3442	0.1334	0.1331	0.2569	0.1753	0.1706	0.2227	0.1932
Embed(0.4)	0.1577	0.0005	0.0010	0.2826	0.0001	0.0003	0.3485	0.0001	0.0001
Embed(0.4)+Citation	0.0600	0.3702	0.1032	0.0951	0.2610	0.1394	0.1150	0.2281	0.1529
Embed(0.4)+BM25	0.0751	0.3663	0.1246	0.1181	0.2578	0.1620	0.1414	0.2235	0.1732
Embed(0.4)+BERT	0.1563	0.0457	0.0708	0.2804	0.0170	0.0320	0.3459	0.0101	0.0197
Embed(0.4)+BERT+BM25	0.0751	0.3664	0.1246	0.1181	0.2594	0.1623	0.1414	0.2286	0.1748
Embed(0.4)+BM25+Citation	0.0600	0.3702	0.1032	0.0951	0.2610	0.1394	0.1150	0.2281	0.1529
Embed(0.4)+BERT+Citation	0.0600	0.3702	0.1032	0.0951	0.2626	0.1397	0.1150	0.2334	0.1541
Embed(0.4)+All	0.0600	0.3702	0.1032	0.0951	0.2626	0.1397	0.1150	0.2334	0.1541
Embed(0.5)	0.0921	0.0016	0.0031	0.1311	0.0003	0.0006	0.1418	0.0001	0.0003
Embed(0.5)+Citation	0.0380	0.4917	0.0706	0.0485	0.3297	0.0845	0.0519	0.3005	0.0885
Embed(0.5)+BM25	0.0486	0.4902	0.0884	0.0606	0.3252	0.1021	0.0642	0.2938	0.1054
Embed(0.5)+BERT	0.0913	0.0978	0.0944	0.1301	0.0253	0.0424	0.1408	0.0149	0.0270
Embed(0.5)+BERT+BM25	0.0486	0.4902	0.0884	0.0606	0.3256	0.1021	0.0642	0.2961	0.1055
Embed(0.5)+BM25+Citation	0.0380	0.4917	0.0706	0.0485	0.3297	0.0845	0.0519	0.3005	0.0885
Embed(0.5)+BERT+Citation	0.0380	0.4917	0.0706	0.0485	0.3303	0.0845	0.0519	0.3029	0.0886
Embed(0.5)+All	0.0380	0.4917	0.0706	0.0485	0.3303	0.0845	0.0519	0.3029	0.0886
Embed(0.6)	0.0681	0.0096	0.0169	0.0700	0.0033	0.0064	0.0704	0.0020	0.0039
Embed(0.6)+Citation	0.0319	0.6364	0.0607	0.0324	0.6009	0.0615	0.0325	0.5969	0.0616
Embed(0.6)+BM25	0.0411	0.6323	0.0772	0.0416	0.5961	0.0778	0.0417	0.5919	0.0780
Embed(0.6)+BERT	0.0675	0.3122	0.1110	0.0694	0.1573	0.0963	0.0698	0.1192	0.0880
Embed(0.6)+BERT+BM25	0.0411	0.6323	0.0772	0.0416	0.5961	0.0778	0.0417	0.5919	0.0780
Embed(0.6)+BM25+Citation	0.0319	0.6364	0.0607	0.0324	0.6009	0.0615	0.0325	0.5969	0.0616
Embed(0.6)+BERT+Citation	0.0319	0.6364	0.0607	0.0324	0.6009	0.0615	0.0325	0.5969	0.0616
Embed(0.6)+All	0.0319	0.6364	0.0607	0.0324	0.6009	0.0615	0.0325	0.5969	0.0616
Embed(0.7)	0.0668	0.0122	0.0206	0.0668	0.0121	0.0205	0.0668	0.0121	0.0205
Embed(0.7)+Citation	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600
Embed(0.7)+BM25	0.0405	0.6355	0.0762	0.0405	0.6355	0.0762	0.0405	0.6355	0.0762
Embed(0.7)+BERT	0.0663	0.3640	0.1122	0.0663	0.3637	0.1121	0.0663	0.3637	0.1121
Embed(0.7)+BERT+BM25	0.0405	0.6355	0.0762	0.0405	0.6355	0.0762	0.0405	0.6355	0.0762
Embed(0.7)+BM25+Citation	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600
Embed(0.7)+BERT+Citation	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600
Embed(0.7)+All	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600	0.0315	0.6404	0.0600

References

1. Altaf, B., Akujuobi, U., Yu, L., Zhang, X.: Dataset recommendation via variational graph autoencoder. In: IEEE International Conference on Data Mining (ICDM), pp. 11–20 (2019)
2. Borgman, C.: One scientist's data as another's noise. Nature **520**(7546), 157 (2015)
3. Brickley, D., Burgess, M., Noy, N.: Google dataset search: building a search engine for datasets in an open web ecosystem. In: WWW Conference, WWW 2019, pp. 1365–1375. ACM (2019). https://doi.org/10.1145/3308558.3313685
4. Chapman, A., et al.: Dataset search: a survey. VLDB J. **29**, 251–272 (2019). https://doi.org/10.1007/s00778-019-00564-x
5. Chen, Y., Wang, Y., Zhang, Y., Pu, J., Zhang, X.: Amender: an attentive and aggregate multi-layered network for dataset recommendation. In: IEEE International Conference on Data Mining (ICDM), pp. 988–993. IEEE (2019)
6. Chinchor, N.: MUC-4 evaluation metrics. In: Proceedings of the 4th Conference on Message Understanding, MUC4 1992, pp. 22–29. ACL (1992). https://doi.org/10.3115/1072064.1072067
7. Cochez, M., Ristoski, P., Ponzetto, S.P., Paulheim, H.: Global RDF vector space embeddings. In: d'Amato, C., et al. (eds.) ISWC 2017. LNCS, vol. 10587, pp. 190–207. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68288-4_12
8. Daza, D., Cochez, M., Groth, P.: Inductive entity representations from text via link prediction. In: Proceedings of The Web Conference (2021). https://doi.org/10.1145/3442381.3450141
9. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT, vol. 1, pp. 4171–4186. ACL, June 2019. https://doi.org/10.18653/v1/N19-1423
10. Duke, C.S., Porter, J.H.: The ethics of data sharing and reuse in biology. BioScience **63**(6), 483–489 (2013)
11. Faniel, I.M., Jacobsen, T.E.: Reusing scientific data: how earthquake engineering researchers assess the reusability of colleagues' data. Comput. Supported Coop. Work **19**(3–4), 355–375 (2010). https://doi.org/10.1007/s10606-010-9117-8
12. Faniel, I.M., Kriesberg, A., Yakel, E.: Social scientists' satisfaction with data reuse. J. Assoc. Inf. Sci. Technol. **67**(6), 1404–1416 (2016)
13. Färber, M.: The microsoft academic knowledge graph: a linked data source with 8 billion triples of scholarly data. In: Ghidini, C., et al. (eds.) ISWC 2019. LNCS, vol. 11779, pp. 113–129. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30796-7_8
14. Färber, M., Leisinger, A.K.: Recommending datasets for scientific problem descriptions. In: International Conference on Information & Knowledge Management, p. 3014 (2021)
15. Fernández, J.D., Martínez-Prieto, M.A., Gutiérrez, C., Polleres, A., Arias, M.: Binary RDF representation for publication and exchange (HDT). Web Semant. Sci. Serv. Agents World Wide Web **19**, 22–41 (2013). http://www.websemanticsjournal.org/index.php/ps/article/view/328
16. Galkin, M., Wu, J., Denis, E., Hamilton, W.L.: NodePiece: compositional and parameter-efficient representations of large knowledge graphs. arXiv preprint arXiv:2106.12144 (2021)
17. Generale, A., Blume, T., Cochez, M.: Scaling R-GCN training with graph summarization (2022). https://doi.org/10.1145/3487553.3524719

18. Martínez-Prieto, M.A., Arias Gallego, M., Fernández, J.D.: Exchange and consumption of huge RDF data. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 437–452. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30284-8_36
19. Pasquetto, I.V., Randles, B.M., Borgman, C.L.: On the reuse of scientific data. Data Sci. J. **16**, 8 (2017)
20. Patra, B.G., Roberts, K., Wu, H.: A content-based dataset recommendation system for researchers-a case study on gene expression omnibus (geo) repository. Database **2020**, 1 (2020)
21. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. ACL (2014). https://doi.org/10.3115/v1/D14-1162
22. Pierce, H.H., Dev, A., Statham, E., Bierer, B.E.: Credit data generators for data reuse (2019)
23. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using Siamese BERT-networks. In: Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992. ACL (2019). https://doi.org/10.18653/v1/D19-1410
24. Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M.M., Gatford, M.: Okapi at TREC-3. In: Overview of the 3rd Text REtrieval Conference (TREC-3), pp. 109–126 (1995). https://www.microsoft.com/en-us/research/publication/okapi-at-trec-3/
25. Tenopir, C., et al.: Changes in data sharing and data reuse practices and perceptions among scientists worldwide. PLOS ONE **10**(8), 1–24 (2015). https://doi.org/10.1371/journal.pone.0134826
26. Wang, X., van Harmelen, F., Huang, Z.: Recommending scientific datasets using author networks in ensemble methods (2022). https://datasciencehub.net/paper/recommending-scienti%EF%AC%81c-datasets-using-author-networks-ensemble-methods
27. Wang, X., van Harmelen, F., Huang, Z.: Biomedical dataset recommendation. In: International Conference on Data Science, Technology and Applications - DATA, pp. 192–199 (2021). https://doi.org/10.5220/0010521801920199
28. Wilkinson, M.D., et al.: The FAIR guiding principles for scientific data management and stewardship. Sci. Data **3**(1), 1–9 (2016)